

1  
2  
3  
4  
5  
6  
7  
8  
9  
0  
1  
2  
3  
4  
5  
6  
7  
8  
9  
0  
1  
2  
3  
4  
5  
6  
7  
8  
9  
0

2  
3  
4  
5  
6  
7  
8  
9  
0  
1  
2  
3  
4  
5  
6  
7  
8  
9  
0  
1  
2  
3  
4  
5  
6  
7  
8  
9  
0

4  
5  
6  
7  
8  
9  
0  
1  
2  
3  
4  
5  
6  
7  
8  
9  
0  
1  
2  
3  
4  
5  
6  
7  
8  
9  
0

5  
6  
7  
8  
9  
0  
1  
2  
3  
4  
5  
6  
7  
8  
9  
0  
1  
2  
3  
4  
5  
6  
7  
8  
9  
0

2  
3  
4  
5  
6  
7  
8  
9  
0

1 The system includes an interface module that comprises an API, which  
2 enables developers of various types of applications, including word processing,  
3 spreadsheet, and presentation design programs, to **include support for the**  
4 **acquisition of images** from various scanners, digital cameras, and image databases.  
(Emphasis added, specification, page 6, lines 29-33.)

5 The image is acquired or provided by an active image acquisition device,  
6 which can be a default device or one selected from a list of available devices  
7 generated by the system, and inserted into the application program document so that  
8 when the document is saved to a file, the captured image comprises a portion of the  
9 file. Furthermore, although it may be temporarily saved into a buffer, the **captured**  
10 **image is never saved to a permanent file** – that is a file that persists after the  
11 application program is closed or the computer is shut down during the process that  
12 inserts the image into document produced by the application program. (Emphasis  
13 added, specification, page 3, lines 18-26.)

14 (See also page 3, line 33 through page 4, line 2, page 29, lines 30-32, and page 47, lines 17-26.)

15 Thus, the application program controls acquisition and communication of the image data directly  
16 from the image acquisition device to the computer memory storing the document. A separate  
17 image acquisition program is not used, and the image data are not first stored in a permanent file  
18 before being inserted into the document in computer memory.

19 Similarly, as to independent Claim 18, applicants have clarified that the application  
20 program used to create the document also enables the user to obtain multiple images directly  
21 from the image acquisition device and insert the multiple images into the document in a  
22 compressed format. Again, the images are inserted into the document without first saving the  
23 compressed format image data to any permanent file. Support for inserting compressed images  
24 is found in the specification at page 3, line 30 through page 4, line 2, which states:

25 The image acquisition device outputs image data corresponding to the  
26 captured image, which is then preferably converted into a compressed format,  
27 preferably using a “lossy” compression scheme known as the Joint Photographic  
28 Experts’ Group (JPEG) format. The compressed image data are then inserted as an  
29 image into an application program document without requiring the user to separately  
30 save the image as a file and then insert the image file into the document.

31 Compressing and inserting multiple images is also discussed in the specification at page 4,  
32 line 33 through page 5, line 1:

1 The user selects the plurality of images to be inserted, and the image data for each  
2 selected image is compressed. The compressed image data are then inserted into the  
3 application program as a plurality of images.

4 In more detail, with regard to independent Claim 24, applicants have clarified that the  
5 interface module is under control of the application program and inserts the data representing the  
6 image into the document stored in the computer memory without saving the data to any  
7 permanent file prior to inserting the data into the document. Support is found in the specification  
8 at page 3, lines 6-12 page 5, lines 17-22, and the discussion of FIGURE 9, which describes a  
9 preferred embodiment of the interface module as an application program interface (API) that is  
10 incorporated into the application program to communicate with the operating system and device  
11 drivers of the image acquisition devices. In particular, "the interface module comprises an API  
12 that allows the application program to easily obtain image data from image acquisition devices  
13 and insert such data into an application program document as one or more images."  
14 (Specification, page 5, lines 17-19).

15 Sobol and Hawes do not disclose, suggest, or provide any motivation for all of the  
16 elements of applicants' amended claims. For example, Sobol and Hawes do not disclose or  
17 suggest acquiring an image using an image source device that is under control of the application  
18 program used to create the document. The Examiner continues to contend that Sobol discloses  
19 acquisition and insertion of a scanned image, with the aid of an active scanner, into a document  
20 using a document creation application. (Office Action, page 2, section 4). However, the  
21 Examiner cites a portion of Sobol that states: "When such line art drawings are scanned, it is  
22 usually with the desire to import or integrate the line-art drawings into other kinds of document,  
23 such as reports or presentation, which themselves may be created with the aid of *separate*  
24 application programs, such as work processors or desktop publishing programs" (emphasis  
25 added, Sobol, col. 2, lines 1-6). Another cited portion of Sobol states that "In one preferred  
26 embodiment, the steps of the method 10 are implemented by an image handler 18 (FIG. 2),  
27 which itself is implemented through a *separate* object linking and embedding program or  
28 system . . ." (emphasis added, Sobol, col. 4, lines 6-10). Similarly, Hawes does not disclose or  
29 suggest any means of acquiring an image. Thus, Sobol and Hawes do not disclose or suggest all  
30 of the elements/steps recited in the amended independent claims.

1 The Examiner also concedes that "Sobol fails to explicitly disclose: *inserting data*  
2 *representing said image . . . all without saving said data in other than a temporary buffer*" (Office  
3 Action, page 2, section 4). Instead, the Examiner continues to argue that Hawes teaches the elements  
4 of inserting an image, and storing the image in a cache or temporary buffer. Applicants again  
5 respectfully disagree and point out that even if Hawes teaches storing an image in a cache and then  
6 inserting the image from the cache, Hawes and Sobol do not disclose or suggest transferring an  
7 image directly from an image acquisition device into a document created by the application program  
8 that is stored in a computer memory, without first saving the image to permanent storage at some  
9 time between acquiring the image from the image acquisition device and inserting the image data  
10 into a document. Applicants' amended claims make it clear that an intervening application program  
11 is not required to communicate with the hardware image acquisition device, and that an intermediate  
12 save operation of the image data to a permanent file is not required.

13 Specifically, as explained above, Sobol acquires images with an image handler that is  
14 separate from the application program used to create/edit a text content of a document. Although  
15 the image handler may be launched from within the application program (See Sobol, col. 6,  
16 lines 31-40), Sobol specifies that the image handler "is a *separate* object linking and embedding  
17 program or system, such as OLE . . ." (emphasis added, Sobol, col. 6, lines 12-14). Sobol further  
18 explains that "Alternatively, however, the method may be implemented as a stand-alone software  
19 package, or by way of any other means known in the art for implementing objects or parts having  
20 distinct attributes or behaviors" (Sobol, col. 6, lines 16-20). Thus, the image handler of Sobol is  
21 a distinct and separate application that controls communication with, and provides settings for,  
22 the hardware image acquisition device.

23 ///

24 ///

25 ///

26 ///

27 ///

28 ///

29 ///

30 ///

1 Hawes uses a browser to reload and display a Web page. However, a browser is not an  
2 application program used to create a text content of a document. A browser is also not an  
3 application program that controls communication with the hardware image acquisition device.  
4 Hawes defines a browser as "a computer program which enables a user to look at information on  
5 other computers and retrieve information from other computers in an environment that allows for  
6 cross-platform communications" (Hawes, col. 1, lines 58-62). Thus, a browser cannot be  
7 interpreted as an equivalent to an application program that creates/edits text, and is not provide  
8 means for communicating with an image acquisition device to transfer image data to the  
9 computer memory comprising a document, as recited in applicants' amended claims.

10 The only element that is potentially relevant to applicants' invention in Hawes is the  
11 concept of using a cache to store data. However, even if the cache from Hawes is combined with  
12 the application program of Sobol, the combination still employs an additional separate image  
13 handler program that is not required by applicants' invention. If the separate image handler were  
14 removed from Sobol, Sobol would not function at all. Thus, a combination of Sobol and Hawes  
15 does not achieve the invention as defined in applicants' amended claims.

16 As to the source driver and source management modules recited in independent Claim 24,  
17 applicants' specification explains that these modules enable an application program to  
18 communicate with an image acquisition device and to control transfer of image data to the  
19 computer memory comprising the document having text content that is created or edited using  
20 the application program. The specification further explains that:

21  
22 The [interface] module [] handles the necessary changes to the application's  
23 event loop, . . . without requiring the event loop within the application to be altered.  
24 Furthermore, the [interface] module provides functionality that is not directly  
25 available from the TWAIN.DLL API module, such as generating drop-down menu  
26 structures comprising a list of available image acquisition devices and verifying  
whether an image acquisition device can actually support automatic scanning.  
(Specification, page 39, lines 27-33).

27  
28 Sobol's separate image handler cannot be an integral part of the application program  
29 itself, so Sobol's separate image handler cannot enable an application program to control an  
30 image acquisition device, as required by applicants' amended claims. At best, Sobol's

1 application program can *launch* a separate image handler, which controls an image acquisition  
2 device.

3 Accordingly, the rejection of amended independent Claims 1, 18, and 24 under  
4 35 U.S.C. § 103(a) should be withdrawn. Also, since a dependent claim inherently includes all  
5 elements of the independent claim from which the dependent claim depends  
6 (MPEP § 608.01(n)), the dependent claim is patentable for at least the same reasons as its base  
7 independent claim. Therefore, the rejection of dependent Claims 2-6, 9-10, 16, 21, 23, 27, 29,  
8 30, 32, and 33 should also be withdrawn for the reasons noted above, in regard to independent  
9 Claims 1, 18, and 24, as amended.

10 Claims Rejected under 35 U.S.C. § 103(a) Over Sobol, Hawes, and Photoshop

11 Dependent Claims 7, 8, 19, 22, 25, and 31 remain rejected under 35 U.S.C. § 103(a)  
12 as being unpatentable over Sobol in view of Hawes, and further in view of “Mastering  
13 Photoshop 5 for the Web” (1998, pp. 1-10, hereinafter “Photoshop”). Applicants have amended  
14 Claims 6 (from which 7 and 8 depend), 22, and 31 to clarify that image enhancement is  
15 performed from within the application program. Support is found in the specification at page 5,  
16 lines 20-22, which state that “The interface module also provides the ability to perform  
17 postprocessing enhancement of the image data, and for compressing the image data prior to  
18 inserting the image(s).” As discussed above, the interface module is under control of the  
19 application program used to create/edit text of the document. In contrast, Photoshop™ is a  
20 separate program and not integral to an application program that creates/edits text in the  
21 document. Accordingly, the rejection of Claims 7, 8, 22, and 31 should be withdrawn.

22 As to Claim 19, control for selecting the plurality of tiled images still must come from  
23 within the word processing application, as required by amended Claim 18. There is no disclosure  
24 or suggestion that Photoshop™ can be used from within an application program of Sobol.  
25 Accordingly, the rejection of Claim 19 under 35 U.S.C. § 103(a) should also be withdrawn.

26 As to Claim 25, this claim simply states that the application program is a word processing  
27 application. Applicants fail to understand how Photoshop might be construed as equivalent to a  
28 word processing application. Although Photoshop may include some rudimentary ability to add  
29 text to images, Photoshop is an image editing program – not a word processing program.  
30 Accordingly, the rejection of Claim 25 under 35 U.S.C. § 103(a) should also be withdrawn.

1 Claims Rejected under 35 U.S.C. § 103(a) Over Sobol, Hawes, and ADF

2 Dependent Claims 12, 13, 15, and 17 remain rejected under 35 U.S.C. § 103(a) as being  
3 unpatentable over Sobol in view of Hawes, and further in view of "IBM ADF Color Scanner  
4 User's Guide" (7/1997, pp. 14-22 and figs. 1-16, hereinafter "ADF"). Applicants have amended  
5 Claim 12, from which Claims 13, 15, and 17 depend, to clarify that the determination is  
6 performed from within the application program. Support is found in the specification at page 43,  
7 lines 6-8, which stat that:

8 [T]he application may desire to bypass the Source's user interface. As  
9 discussed above, this is a necessary capability for performing an autoscan.

10 ADF describes a separate Action Manager program that must be run to use a scanner. More  
11 specifically, the Examiner cites a Figure 2 in the reference, which appears to be a help screen that  
12 indicates that a user may create "actions" that free the user from setting up the same scan job  
13 over and over again. However, even if a user predefined a set of settings in an "action" interface,  
14 the user must at least select the "action" (i.e., which set of settings) desired before the  
15 corresponding settings are automatically used. For instance ADF explains that "The Action  
16 Manager displays every time you insert a document into the scanner. This gives you a chance to  
17 change actions or to select another Windows application' (ADF, page 22, lines 3-5). Thus, ADF  
18 does not disclose or suggest determining automatic image scan without requiring a user to select  
19 image capture parameters from within an application used to create or edit text in a document.  
20 Accordingly, the rejection under 35 U.S.C. § 103(a) of dependent Claim 12 should be  
21 withdrawn. Because Claims 13, 15, and 17 depend from Claim 12, the rejection of  
22 Claims 13, 15, and 17 should be withdrawn for at least the same reasons as Claim 12.

23 ///

24 ///

25 ///

26 ///

27 ///

28 ///

29 ///

30 ///

1 Also, as discussed above, the rejection of other dependent claims not specifically  
2 addressed should be withdrawn for the reasons above regarding the independent claims from  
3 which the dependent claims depend. In view of the preceding amendments and remarks, it will  
4 be apparent that all claims in this case currently define a novel and non-obvious invention, and  
5 that the application is in condition for allowance and should be passed to issue without further  
6 delay. Should any further questions remain, the Examiner is invited to telephone applicants'  
7 attorney at the number listed below.

8  
9 Respectfully submitted,

10 

11  
12 Ronald M. Anderson  
13 Registration No. 28,829

14 RMA/TRM:

15 I hereby certify that this correspondence is being deposited with the U.S. Postal Service in a  
16 sealed envelope as first class mail with postage thereon fully prepaid addressed to: Director of Patents  
and Trademarks, Arlington, VA 22202, on March 31, 2003.

17 Date: March 31, 2003

18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30 